

AMLS GRANT - 1N-61

76826 - CR

119

Research on Knowledge Representation, Machine Learning, and Knowledge Acquisition

Final Report, covering the period 10/1/83 - 1/31/87

NASA Grant Number NCC 2-274

Bruce G. Buchanan, Principal Investigator

23 June 1987

Knowledge Systems Laboratory
Computer Science Department
Stanford University
Stanford, CA 94305

(NASA-CR-180408) RESEARCH ON KNOWLEDGE REPRESENTATION, MACHINE LEARNING, AND KNOWLEDGE ACQUISITION Final Report, 1 Oct. 1983 - 31 Jan. 1987 (Stanford Univ.) 11 p
Avail: NTIS HC A02/MF A01 CSCL 09B G3/61 0076826
N87-24103
Unclas

This report summarizes the research work performed at the Knowledge Systems Laboratory of Stanford University under NASA Contract NCC 2-274 between October 1, 1983 and January 31, 1987.

Research on Knowledge Representation, Machine Learning, and Knowledge Acquisition

Final Report, covering the period 10/1/83 - 1/31/87

NASA Grant Number NCC 2-274

This report summarizes research in knowledge representation, machine learning, and knowledge acquisition performed at the Knowledge Systems Laboratory (KSL) and supported by NASA Cooperative Agreement number NCC 2-274. This work was performed over a period of more than three years, beginning in October of 1983 and ending in January 1987. The research conducted under this contract is continuing under NASA Contract NCC 2-220, research on architectures for concurrent symbolic computation, also at the KSL. Professors Edward A. Feigenbaum and Bruce G. Buchanan are co-Principal Investigators for this combined effort, which is entitled "Cooperating Knowledge-Based Systems".

The goals of the research under NCC 2-274 are outlined below, followed by a summary of research progress over the three years. All technical publications of the Knowledge Systems Laboratory that are referenced in this report can be obtained by writing to the KSL.

Goals of the Research

Knowledge Representation and Use

The first major goal of this research is to develop flexible, effective methods for representing the qualitative knowledge necessary for solving large problems that require symbolic reasoning as well as numerical computation. Representing knowledge for computers entails finding a set of conventions for describing facts and relations about a problem that computers can use effectively. Over the last 25 years, research in artificial intelligence (AI) has produced several techniques for successfully representing and utilizing qualitative, non-mathematical information, including semantic nets, frames, logic, rules, and procedures. In particular, our research focuses on integrating different representation methods to describe different kinds of knowledge more effectively than any one method alone can. *

In particular, we have focused on representing and using spatial information about three-dimensional objects and constraints on the arrangement of these objects in space. A computer system for reasoning about spatial relationships must have flexible and powerful capabilities to describe objects and constraints at several levels of abstraction, to include qualitative as well as numeric constraints, to define procedures that operate on objects and apply constraints to find desirable configurations of objects, and to represent heuristics that guide the efficient application of these procedures.

We have chosen an application domain for spatial reasoning and knowledge representation that is closely allied with many tasks of assembling complex three-dimensional structures: the assembly of a protein molecule from its atomic constituent parts. In this application, as in many spatial problem domains, the structure to be built must satisfy numerous constraints, only some of which are expressible as numeric limits. The program must understand alternative subassemblies and several levels of abstraction of the structure, and must reason about the constraints, objects, and operations that position the objects in order to solve the structure in reasonable amount of time. In addition, the system faces the possibility of inconsistent and noisy information in its data.

Our work on this problem has resulted in a prototype expert system for protein structure determination called PROTEAN. It is implemented in BB1, a blackboard architecture that has facilities for representing the structures, relationships, and procedures needed to construct three-dimensional objects. This system architecture provides facilities for representing and integrating diverse kinds of knowledge, including problem-solving strategies (control knowledge) as well as domain specific knowledge. Progress on BB1 and PROTEAN are included in the following sections.

In the last year of this research contract, we have also focused on a second application in Financial Resource Management (FRM). FRM assists in judgmental aspects of budget planning and management of the personnel, time, equipment, and financial resources of an organization. Commonly available tools (spreadsheets and databases) for such tasks provide little assistance in the representation and manipulation of symbolic information such as policies, procedures, and promises that are essential for effective planning and management. This domain presents interesting and challenging AI issues in knowledge acquisition, knowledge representation, constraint satisfaction and heuristic planning. Recent work on the FRM system is described below.

Machine Learning and Knowledge Acquisition

A second major theme, included in the final year of this contract, is the development of robust machine learning programs that can be integrated with a variety of intelligent systems. To make effective use these programs, it is also necessary to define criteria under which machine learning techniques can be successfully applied to different problem-solving architectures.

To achieve this goal, we are designing, implementing, and experimenting with learning methods in several different problem-solving environments. This work involves developing methods to learn strategic (or control) heuristics in the course of problem solving, developing tools for creating knowledge sources and knowledge representation, and examining the role of noise, the use of examples and counterexamples, and methods of knowledge representation in machine learning.

Our research in machine learning has focused on several distinct problem domains including medical (NEOMYCIN/HERACLES) and biochemical (PROTEAN) in addition to domain-independent investigations. We also are motivated by the need for effective tools for knowledge acquisition and maintenance of knowledge bases (IMPULSE in STROBE, and BBEDIT and KSEDIT in BB1).

Research Progress

PROTEAN Progress

PROTEAN is an evolving knowledge-based system that determines the three-dimensional structure of protein molecules. The program uses empirically determined constraints as data and expert biochemical knowledge of protein structure and behavior to analyze this data and derive solutions to the structure. The problem is important, not only to chemists and biologists interested in the detailed results of the protein geometry, but also for the knowledge representation methods, problem-solving techniques, and heuristic approaches that are being developed for the assembly of structures subject to complex constraints.

Substantial progress was made on the PROTEAN in the first two years of this contract. First, a conceptual framework for representing protein structure at several levels of detail was designed. Second, specific actions for positioning three-dimensional objects subject to constraints were defined. Third, BB1 knowledge sources were built to implement the blackboard effects of these actions in the reasoning component. Fourth, two prototypes of a geometric constraint satisfaction system were built to actually compute the results of these actions, one implemented in LISP and the other in the C programming language. Fifth, we developed a graphics display program to show the results of the geometric computations in three dimensions on a Silicon Graphics IRIS graphics terminal.

Three technical papers describe our initial approach and preliminary results from this work. The first two of these are discussions of the protein structure determination problem, oriented toward a biochemical audiences [20, 21]. The third report [7] includes a more technical description of the AI approach and methods used in PROTEAN to solve the structure of a small protein at the "solid" level of abstraction, in which secondary structures of the protein are represented as simple geometric solids.

PROTEAN is currently implemented as a distributed computation system, with the reasoning component running on a Xerox 1100 LISP workstation in BB1 (see below) and the geometric

and display components running on an IRIS graphics workstation. These machines communicate instructions and results of their computations over a local area network.

In August 1986, Dr. Barbara Hayes-Roth presented a paper on PROTEAN to the 1986 AAAI conference [17]. Altman and Jardetzky presented the PROTEAN approach to protein structure using empirical data without consideration of theoretical constraints in [1]. Lichtarge presented a systematic characterization and validation of the PROTEAN method in his Ph.D. thesis [23] and, with other authors, in [24].

In 1986, the initial prototype of the geometry system was refined to increase its generality and improve its ability to represent many kinds of structures. In [3, 4], Brinkley et al. describe this general system as employed by PROTEAN to determine possible locations for objects. The placement of objects subject to constraints is a combinatorially explosive task, and it is the core problem that PROTEAN solves. PROTEAN attempts to make the problem computationally feasible by refining a solution in three ways: (a) solving the problem at several levels of detail; (b) using constraint satisfaction algorithms to reduce the number of possible solutions to enumerate; and (c) employing heuristics to choose the order in which constraints are applied.

The computations of the system described above are time consuming and expensive to perform. The time needed to solve a structure depends crucially on the order in which geometric computations are performed, suggesting that intelligent selection of actions would be useful to increase the efficiency of PROTEAN. Garvey et al. [13] investigated several control strategies in PROTEAN to determine the cost of control knowledge compared with the benefits of using it. They found that different kinds of control knowledge have different costs and degrees of effectiveness, but that the cost of additional control was generally less than the benefit of improved efficiency. In addition, these relative benefits actually increase with the complexity and computational requirements of the problem.

Altman and Buchanan [2] explore the utility of compiled knowledge in the construction of protein structures as a way to increase the efficiency of part of the problem-solving activity of PROTEAN. Their approach begins with a declarative representation of control strategies and partially *compiles* this knowledge into procedures that plan the long-range strategy of the system. They find that a combination of methods that take advantage of unexpected opportunities in the evolving solution ("opportunistic" knowledge sources) and a procedurally defined global plan retains much of the flexibility of a declarative representation of control while gaining advantages learned from previous experience with the PROTEAN system.

PROTEAN research is continuing, focused on two aspects. First, research on representation and computation at the atomic level of detail is making a more detailed description of protein structure available. Second, we continue to experiment with strategies for more efficient assembly of the protein in one subunit, and for building and combining subunits of the protein in a "divide and conquer" approach.

BB1 Progress

BB1 is a knowledge based system using a blackboard architecture for control, described in [16]. It is the system in which the reasoning components of PROTEAN are implemented, chosen because of its ability to integrate diverse kinds of data and independent sources of expertise. BB1 was first implemented in 1984 and 1985 in Interlisp on a large DEC-20 computer and Xerox 1100 series LISP workstations. Since that time, the system has been translated into CommonLISP and is now available on a number of other computers [11, 12].

A knowledge source (KS) is a source of expertise on a part of the problem-solving task. BB1 can include three different kinds of knowledge sources:

- a domain KS specifies actions that directly contribute to the evolving solution on the blackboard,
- a control KS contains strategic knowledge about which of several possible actions is the best to take during the problem-solving,
- a learning KS observes changes on the blackboard and the behavior of an expert using the system to create new strategic or domain knowledge.

The BB1 architecture differs from other blackboard systems by defining structures for the support of control KSs and strategic knowledge.

To assist in the development of control knowledge sources, the BB1 framework was enhanced to include MARCK, a module that builds control knowledge sources interactively during the execution of BB1 [15]. The system asks the user at each step if the action that BB1 rates most highly is the best one to take. If the expert indicates that another action is more desirable, MARCK is activated and interviews the expert to determine how BB1's choice differs from the user's choice. MARCK then uses this information to create a heuristic and programs it in a control knowledge source. This heuristic is then available for further cycles of the expert system and may improve the problem-solving performance.

Improvements are continually being made to the BB1 framework. During the last year, the versatility and usefulness of the BB1 system have been enhanced by several developments described below.

BBEDIT is a tool built by Alan Garvey for creating and maintain knowledge bases for use with BB1. It has encouraged the development of several versatile, independent knowledge bases describing biochemical and problem solving concepts, and has encouraged formalization and specification of knowledge formerly in procedures in the PROTEAN system. The independence of the knowledge bases allows their use in related expert systems.

KSEDIT is a specialized editor for control and domain knowledge sources in BB1, built by Micheal Hewett. KSEDIT facilitates the building of syntactically correct knowledge sources and allows the user to focus on the logical structure of the system rather than on the detailed syntax of the knowledge source.

Hayes-Roth et al. [18] developed **ACCORD**, a layered environment for reasoning about problem-solving actions in the class of arrangement-assembly problems, in which solutions are created by arranging objects by assembly. The ACCORD framework is a set of knowledge structures used to represent actions, events, states, and facts involved in solving problems by the method of construction subject to constraints. ACCORD is used in PROTEAN, but is applicable to many varied tasks including construction site layout and travel planning.

ExAct is a module for explaining the actions of a system in BB1. In [26], Schulman describes the explanation facility of BB1 that describes actions along with the considerations and decisions that lead up to them. ExAct takes advantage of the structure of the ACCORD language to explain system actions and ratings in terms of the heuristics and control plans on the control blackboard.

Goal-directed reasoning has been demonstrated in BB1 in PROTEAN, augmenting the existing hierarchical planning capabilities. A paper by Johnson [22] describes the simultaneous use and integration of hierarchical, opportunistic, and goal-directed strategies to determine protein structure. The reasoning mechanism exploits BB1's control semantics (actions, events) and ACCORD's representation of the relations between actions, events, and states to deliberately promote particular kinds of actions and detect opportunities to perform generally desirable actions.

BB1 is an AI architecture that has been fully implemented in both Interlisp-D and in CommonLISP. In the past, very little has been published on the development and implementation of AI systems. In [19], Hewett discusses the software architecture of BB1 and reviews the design decisions and their consequences in this implementation.

FRM - Financial Resource Management

The Financial Resource Management (FRM) system is discussed in a paper by Gelman et al. [14]. FRM assists in judgmental aspects of budget planning and management of the personnel, time, equipment, and financial resources of an organization. Commonly available tools (spreadsheets and databases) for such tasks provide little assistance in the representation and manipulation of symbolic information such as policies, procedures, and promises that are essential for effective planning and management.

Capturing the domain knowledge of expert managers is crucial, since much of the expertise of effective managers is acquired only by experience, and is neither documented nor passed

from person to person. From an AI perspective, this domain presents many complex constraint satisfaction problems. FRM represents constraints in a knowledge base, including priority and context information, and applies them to budget preparation and budget replanning while presenting information to the user in a spreadsheet-like format.

Inherent in FRM is the need for a "smart" user interface, to enable the user to express constraints easily and naturally as well as to see the results of the application of these constraints in a familiar format. FRM is currently implemented using STROBE [28], an object-oriented programming environment.¹ IMPULSE, a powerful editor for STROBE systems [25], allows extensive customization and is used in FRM as a "smart" editor for constraints.

Progress in Machine Learning

Buchanan et al. present an empirical study of the incremental learning process using a careful selection of counter examples in concept formation with the rule-learning system RL [9]. They find that "near misses", negative examples that are similar to acceptable cases, are particularly effective in shrinking the space of possible theories that explain the examples observed. They define and use a metric for the distance of each example from the target theory and measure the effectiveness and efficiency of examples related to the distance measured, demonstrating that the power of near misses to restrict the space of possible theories results from their small distance from the target. They also find that intelligent selection of instances based upon knowledge of the state of the evolving theory results in a faster convergence of an evolving theory toward the target concept, requiring many fewer cases for learning.

Debugging Knowledge Structures

In large rule-based systems, the performance of the system is strongly dependent on the degree to which the knowledge of the system is "debugged" and refined, i.e. erroneous rules are identified and removed, redundant rules are combined, missing rules are added, and certainty factors of rules are found that give good results over many cases. Such evaluation and restructuring of knowledge is an important type of learning and can be automated to some extent. Here we describe recent work in the debugging and refinement of knowledge bases using several techniques.

Wilkins and Buchanan [30] describe a problem with the rule sets of rule-based systems that use certainty factors, i.e. better individual rules do not necessarily lead to a better overall set of rules. Since all less-than-certain rules contribute evidence towards erroneous conclusions for some problem instances, the distribution of these erroneous conclusions is not necessarily related to the quality of individual rules. This has important consequences for automatic machine learning of rules, since rule selection is usually based on measures of quality of individual rules. The authors present a method using a new Antidote Algorithm that performs a model-directed search of the rule space to find an improved rule set. They report that the application of this method significantly reduced the number of misdiagnoses when applied to a rule set generated from 104 training instances.

Debugging the knowledge structures of a problem solving agent is discussed using the *synthetic agent* method [31], to determine a *performance upper bound* for debugging a knowledge base. The synthetic agent systematically explores the space of near miss training instances and expresses the limits of debugging in terms of the knowledge representation and control language constructs of the expert system. This paper presents the framework for evaluating a differential modeling system.

In [32], Wilkins describes the ODYSSEUS apprenticeship learning program, designed to refine and debug knowledge bases for the HERACLES expert system shell. ODYSSEUS analyzes the behavior of a human specialist using two underlying domain theories, a *strategy*

¹STROBE, like KEE and LOOPS, is a descendant of the UNIT package [27] developed for the MOLGEN project at Stanford University [10].

theory for the problem solving method (heuristic classification), and an *inductive theory* based on past problem solving sessions. ODYSSEUS improves the knowledge base for the expert system shell, identifying bugs in the system's knowledge in the process of following the line-of-reasoning of an expert, serving as a knowledge acquisition subsystem. The system can also be used as part of an intelligent tutor, identifying problems in a novice's understanding and serving as student modeler for tutoring systems.

Wilkins et al. illustrate that an explicit representation of the problem solving method and underlying theories of the problem domain provide a powerful basis for automating learning for expert system shells [33]. By using domain-independent *task procedures* and *task procedure metarules*, domain knowledge can be located and applied to achieve problem solving subgoals. However, these rules are often limited in use due to insufficient domain knowledge. This paper describes the use of *metarule critics* in ODYSSEUS for automating the acquisition of domain knowledge, illustrating a powerful form of failure-driven learning at the level of subgoals as well as at the level of solving the entire problem.

Publications of Interest

Buchanan presents a discussion of rule-based expert systems in his article [5]. In this report, he discusses automated reasoning and the heuristic approach, the history and characteristics of expert systems, and the "expertness" of these programs. He also presents some directions for the future of expert systems technology and its application.

In [6], Buchanan presents a list of expert systems that were being actively used in academic and industrial applications at the time of writing. This listing is categorized by application area and, although incomplete and now out of date, indicates the importance of expert systems technology in actual applications.

Subramanian and Buchanan have prepared a reading list for students of artificial intelligence [29]. This technical report is used as a study guide for Ph.D. students for the qualifying examinations at Stanford University, containing references to seminal papers in major fields of AI research.

Buchanan discusses the nature of "Artificial Intelligence as an Experimental Science" [8] and argues that observation and experimentation in this field, as in the physical sciences, will improve our understanding of intelligent behavior of people and computers. He also presents examples of projects that have completed a *research cycle* by analyzing data collected from demonstrations of AI systems and generalizing their results. He suggests that AI research may benefit from the combination of an empirical, experimental approach with careful evaluation and characterization of AI methods and their results.

Other NASA-related Activities

Dr. Craig Cornelius, research associate in the Helix Group of the KSL, presented a talk entitled "PROTEAN: Deriving Protein Structure From Constraints" to the AI Research Forum held at NASA-Ames Research Center July 22-24, 1987. He discussed the objectives of the PROTEAN project, along with the status of BB1 and PROTEAN and plans for extension of the work. A report on this research was also submitted to "NASA Proceedings".

A group of 15 managers from the 8 NASA centers visited the Knowledge Systems Laboratory on April 24, 1987 to explore areas of mutual interest in artificial intelligence research and applications. In an all-day briefing, several KSL researchers presented some aspects of their current work that are relevant to the interests of NASA.

References

- [1] Altman, R., and Jardetzky, O.
New strategies for the determination of macromolecular structures in solution.
J. Biochem. (Tokyo) 100(6):1403-1423, 1986.
- [2] Russ B. Altman and Bruce G. Buchanan.
Partial Compilation of Control Knowledge.
1987.
Submitted to AAAI-87.
- [3] James Brinkley, Craig Cornelius, Russ Altman, Barbara Hayes-Roth, Olivier Lichtarge, Bruce Duncan, Bruce Buchanan, Oleg Jardetzky.
Application of Constraint Satisfaction Techniques to the Determination of Protein Tertiary Structure.
Technical Report KSL 86-28, Stanford University, Knowledge Systems Laboratory, March, 1986.
- [4] James F. Brinkley, Bruce G. Buchanan, Russ B. Altman, Bruce S. Duncan, Craig W. Cornelius.
A Heuristic Refinement Method for Spatial Constraint Satisfaction Problems.
Technical Report KSL 87-05, Stanford University, Knowledge Systems Laboratory, January, 1987.
- [5] Bruce G. Buchanan.
Expert Systems.
Journal of Automated Reasoning 1(1), 1984.
Also Technical Report HPP 84-15.
- [6] Bruce G. Buchanan.
Expert Systems: Working Systems and the Research Literature.
Expert Systems , December, 1985.
Also Technical Report KSL 85-37.
- [7] Bruce G. Buchanan, Russ Altman, James Brinkley, Craig Cornelius, Bruce Duncan, Barbara Hayes-Roth, Micheal Hewett, Olivier Lichtarge, Oleg Jardetzky.
The Heuristic Refinement Method for Deriving Solution Structures of Proteins.
Technical Report KSL 85-41, Knowledge Systems Laboratory, Stanford University, Stanford, CA., March, 1986.
10 pages.
- [8] Bruce G. Buchanan.
Artificial Intelligence as an Experimental Science.
Technical Report KSL 87-03, Stanford University, Knowledge Systems Laboratory, January, 1987.
To appear in Synthese.
- [9] Bruce G. Buchanan, Edwina L. Rissland, Paul S. Rosenbloom, Hwee Tou Ng.
The Role of Intelligent Instance Selection in Learning Systems: the Near Miss.
1987.
In preparation.

- [10] Friedland, P.E.
Knowledge-Based Experiment Design in Molecular Genetics.
PhD thesis, Stanford University, October, 1979.
Also technical report HPP-79-29.
- [11] Alan Garvey, Micheal Hewett, M. Vaughan Johnson, Robert Schulman, Barbara Hayes-Roth.
BB1 User Manual - Interlisp Version.
Technical Report KSL 86-60, Knowledge Systems Laboratory, Stanford University,
October, 1986.
Working Paper.
- [12] Alan Garvey, Micheal Hewett, M. Vaughan Johnson, Robert Schulman, Barbara Hayes-Roth.
BB1 User Manual - Common LISP Version.
Technical Report KSL 86-61, Knowledge Systems Laboratory, Stanford University,
October, 1986.
Working Paper.
- [13] Alan Garvey, Craig Cornelius and Barbara Hayes-Roth.
Computational Costs versus Benefits of Control Reasoning.
Technical Report KSL 87-11, Stanford University, Knowledge Systems Laboratory,
February, 1987.
- [14] A.M. Gelman, T.C. Rindfleisch, C.W. Cornelius, B.G. Buchanan.
FRM: An Intelligent Assistant for Financial Resource Management.
Technical Report KSL-87-17, Stanford University, Knowledge Systems Laboratory,
February, 1987.
- [15] Hayes-Roth, B., and Hewett, M.
Learning Control Heuristics in a Blackboard Environment..
Technical Report HPP 85-2, Knowledge Systems Laboratory, Stanford University,
Stanford, CA., 1985.
- [16] Hayes-Roth, B.
A Blackboard Architecture for Control.
Artificial Intelligence Journal 26:251-321, 1985.
- [17] Hayes-Roth, B., Buchanan, B., Lichtarge, O., Hewett, M., Altman, R., Brinkley, J.,
Cornelius, C., Duncan, B., Jardetzky, O.
PROTEAN: Deriving protein structure from constraints.
In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages
904-909. American Association for Artificial Intelligence, Philadelphia, Penn.,
August 11-15, 1986.
- [18] Barbara Hayes-Roth, Alan Garvey, M. Vaughan Johnson Jr., and Micheal Hewett.
A Layered Environment for Reasoning about Action.
Technical Report KSL 86-38, Stanford University, Knowledge Systems Laboratory,
November, 1986.
- [19] Micheal Hewett and Barbara Hayes-Roth.
The BB1 Architecture: A Software Engineering View.
Technical Report KSL 87-10, Stanford University, Knowledge Systems Laboratory,
February, 1987.

- [20] Jardetzky, O., Lane, A., Lichtarge, O., Hayes-Roth, B., and Buchanan, B.
Determination of macromolecular structure and dynamics by NMR.
In *Proceedings of the NATO Advanced Study Institute "NMR in the Life Sciences"*.
Plenum Publishing Corporation, 1985.
- [21] Jardetzky, O., Lane, A.N.L., Lefevre, J.F., Lichtarge, O., Hayes-Roth, B., Altman, R., and Buchanan, B.
A new method for the determination of protein structures in solution from NMR.
In B. Maraviglia, F. de Luca, R. Campanella (editors), *Proceedings of the XIII Congress Ampere on Magnetic Resonance*, pages 64-69. Congress Ampere, Rome, Italy, 1986.
- [22] M. Vaughan Johnson Jr. and Barbara Hayes-Roth.
Integrating Diverse Reasoning Methods in the BBI Blackboard Control Architecture.
Technical Report KSL 86-76, Stanford University, Knowledge Systems Laboratory,
December, 1986.
- [23] Lichtarge, Olivier.
Structure determination of proteins in solution by NMR.
PhD thesis, Stanford, November, 1986.
- [24] Olivier Lichtarge, Craig W. Cornelius, Bruce G. Buchanan, Oleg Jardetzky.
Validation of the First Step of the Heuristic Refinement Method for the Derivation of
Solution Structures of Proteins from NMR Data.
1986.
- [25] Schoen, E. and Smith, R.G.
IMPULSE, A Display-Oriented Editor for STROBE.
In *Proceedings of the National Conference on Artificial Intelligence*, pages 356-358.
American Association for Artificial Intelligence, Morgan Kaufmann Publishers,
August, 1983.
- [26] Robert Schulman and Barbara Hayes-Roth.
ExAct: A Module for Explaining Actions.
Technical Report KSL 87-8, Stanford University, Knowledge Systems Laboratory,
January, 1987.
- [27] Smith, R.G. and Friedland, P.
UNIT Package User's Guide.
Technical Report HPP-80-28, Stanford University, Heuristic Programming Project,
December, 1980.
- [28] Smith, R.G. and Carando, P.
Structured Object Programming in STROBE.
Technical Report SYS-86-17, Schlumberger-Doll Research, October, 1986.
- [29] Devika Subramanian and Bruce G. Buchanan.
A General Reading List for Artificial Intelligence.
Technical Report KSL 85-54, Knowledge Systems Laboratory, Stanford University,
Stanford, CA., December, 1985.
63 pages.

- [30] David C. Wilkins and Bruce G. Buchanan.
On Debugging Rule Sets When Reasoning Under Uncertainty.
Technical Report KSL 86-30, Stanford University, Knowledge Systems Laboratory, July, 1986.
Earlier version appeared in Proceedings of the Fifth National Conference on Artificial Intelligence, August 1986.
- [31] David C. Wilkins, William J. Clancey and Bruce G. Buchanan.
Using and Evaluating Differential Modeling in Intelligent Tutoring and Apprentice Learning Systems.
Technical Report KSL 86-62, Stanford University, Knowledge Systems Laboratory, October, 1986.
- [32] David C. Wilkins.
Knowledge Base Debugging Using Apprenticeship Learning Techniques.
Technical Report KSL 86-63, Stanford University, Knowledge Systems Laboratory, December, 1986.
- [33] David C. Wilkins, William J. Clancey and Bruce G. Buchanan.
Knowledge Base Refinement Using Abstract Control Knowledge.
Technical Report KSL 87-01, Stanford University, Knowledge Systems Laboratory, January, 1987.